



HTML5 / CSS3 / jQuery / jQuery Mobile / JavaScript
超过300个实用代码范例教学，讲练结合，快速上手

HTML5+ jQuery Mobile 移动应用开发

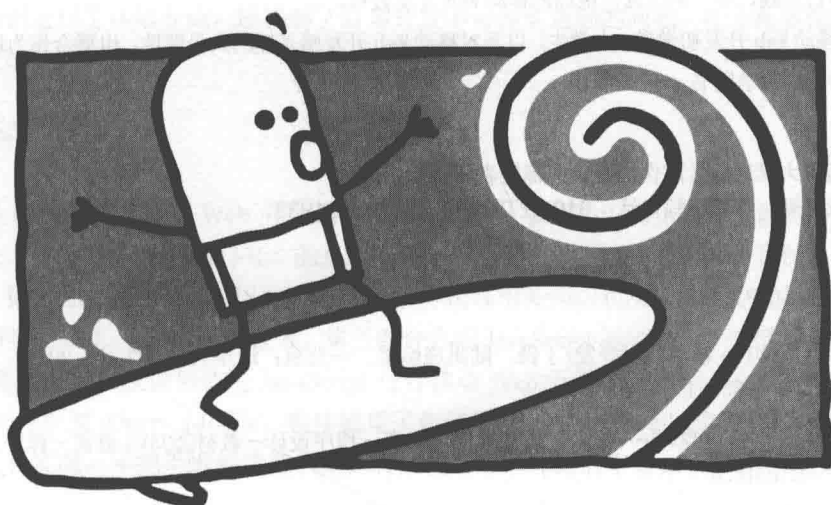
丁 锋 陆禹成 编著

轻松构建移动Web应用



清华大学出版社





HTML5⁺ jQuery Mobile

移动应用开发

丁 锋 陆禹成 编著

清华大学出版社
北 京

内 容 简 介

本书主要对移动 Web 开发进行介绍, 主要内容包括 HTML5 相关技术、界面样式及 CSS3 相关技术、Javascript 基本使用以及 jQuery Mobile 移动开发框架。

本书内容涵盖移动 Web 开发相关的基础知识、开发经验、针对移动端的开发技巧、移动开发框架以及项目实战。书中详细介绍移动 Web 开发的前沿技术, 比对了传统 Web 开发和移动 Web 开发的区别, 对 HTML5、CSS3 等前沿技术进行了详细深入的介绍, 对移动应用开发中的移动设备适配、Web 实时通信等核心问题进行了详细讲解。在理论的基础上, 注重项目实践, 是一本可读性很高的移动 Web 开发教程。

本书适合移动 Web 开发初学者、大学生, 以及对移动 Web 开发感兴趣的人员阅读, 也适合作为培训机构或大专院校及职业学院的教学用书。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

HTML5+jQuery Mobile 移动应用开发/丁锋, 陆禹成编著. —北京: 清华大学出版社, 2018
ISBN 978-7-302-49350-1

I. ①H… II. ①丁… ②陆… III. ①超文本标记语言—程序设计—教材②JAVA 语言—程序设计—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2018)第 014494 号

责任编辑: 王金柱

封面设计: 王 翔

责任校对: 闫秀华

责任印制: 杨 艳

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 北京富博印刷有限公司

装 订 者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 190mm×260mm

印 张: 28.5

字 数: 730 千字

版 次: 2018 年 3 月第 1 版

印 次: 2018 年 3 月第 1 次印刷

印 数: 1~3000

定 价: 79.00 元

产品编号: 069961-01

前言

HTML5、CSS3、jQuery Mobile 框架等知识均为时下移动 Web 开发的前沿技术。本书不仅对基本的开发知识进行了详细介绍，还讲解了许多开发中可能会遇到的问题，在介绍基础知识的同时讲解大量的开发经验。

本书内容

本书从多个角度对移动 Web 开发进行详细介绍，从基础知识到实战开发都有所涉及。本书主要分为五部分，第一部分介绍 HTML5 的基础知识，包括语义化标签、HTML5 存储、实时通信技术、多媒体等。第二部分介绍 CSS3 以及移动端开发样式等相关知识和应用方法，包括移动端元素单位、移动端布局基础知识、CSS3 新特性等。第三部分介绍 JavaScript 基础知识，包括 JavaScript 基本语法、JavaScript 常用方法以及如何使用 JavaScript 进行移动 Web 开发。第四部分介绍非常流行、高效的移动端 Web 开发框架 jQuery Mobile，整体梳理了如何使用 jQuery Mobile 进行移动端 Web 开发。第五部分介绍项目实战，手把手地教读者从无到有，使用本书介绍的丰富知识和经验进行移动 Web 应用开发。

- 第 1 章 认识 HTML
- 第 2 章 语义化标签
- 第 3 章 视频和音频
- 第 4 章 存储
- 第 5 章 即时通信
- 第 6 章 Canvas 绘图
- 第 7 章 CSS 选择器
- 第 8 章 元素定位
- 第 9 章 移动元素单位
- 第 10 章 盒模型
- 第 11 章 Flex 布局——FlexBox
- 第 12 章 使用 CSS3 新特性
- 第 13 章 DOM 操作
- 第 14 章 JavaScript 对象
- 第 15 章 JavaScript 基本语法
- 第 16 章 Window 对象
- 第 17 章 函数
- 第 18 章 jQuery 中的选择器

- 第 19 章 jQuery 中的 DOM 方法
- 第 20 章 jQuery 操纵 CSS 样式
- 第 21 章 jQuery 中的动画效果
- 第 22 章 jQuery 中的 AJAX
- 第 23 章 jQuery Mobile 组件
- 第 24 章 jQuery Mobile 的触摸事件
- 第 25 章 jQuery Mobile 表单
- 第 26 章 jQuery Mobile 列表
- 第 27 章 jQuery Mobile 项目结构
- 第 28 章 项目实战：聊天 APP 的开发

下载资源

本书下载资源地址：<https://pan.baidu.com/s/1bpvw3uf>（注意区分数字和英文字母大小写）。如果下载有问题，请发送电子邮件至 booksaga@126.com，邮件标题为“HTML5+jQuery Mobile 移动应用开发下载资源”。

为了让大家更好地学习 HTML5 的相关知识，还可以加 QQ 技术交流群：425975187。

读者对象

本书适合零基础、想从事移动 Web 前端开发的开发者以及对移动 Web 前端技术感兴趣的读者阅读。无论是在校计算机相关专业学生，还是已经从事移动 Web 开发的开发者，都能在本书中有所收获。

由于作者水平有限，书中难免有疏漏之处，敬请广大读者批评指正。

编 者
2018 年 1 月

目 录

第 1 章 认识 HTML	1	3.3.3 HTML5 中 audio 和 video 的 事件	36
1.1 HTML 基本介绍	1	第 4 章 存储	38
1.2 HTML 常用标签	2	4.1 本地存储	39
1.3 HTML 表单	3	4.2 localStorage 对象	39
1.4 HTML 和 CSS 的简单交互	4	4.2.1 通过 localStorage 设置存储 元素	40
第 2 章 语义化标签	5	4.2.2 获取数据	43
2.1 什么是语义化标签	5	4.2.3 删除数据	44
2.2 header 标签	6	4.2.4 清除数据	45
2.3 footer 标签	8	4.3 sessionStorage	48
2.4 nav 标签	9	4.3.1 存储数据	48
2.5 article 标签	10	4.3.2 获取数据	50
2.6 section 标签	12	4.3.3 删除数据	51
2.7 aside 标签	13	4.3.4 清除数据	51
第 3 章 视频和音频	15	4.4 storage 对象	52
3.1 音频播放器——audio	15	4.5 storage 事件	53
3.1.1 audio 标签的使用	15	4.6 应用程序缓存	54
3.1.2 如何获取音频文件的 URL	16	4.7 本地数据库	57
3.1.3 autoplay 属性	17	4.7.1 创建/打开数据库	57
3.1.4 loop 属性	18	4.7.2 执行操作	58
3.1.5 preload 属性	18	第 5 章 即时通信	61
3.1.6 audio 标签支持的音频文件	18	5.1 概述	61
3.2 视频播放器——video	19	5.1.1 Web 端即时通信技术	61
3.2.1 video 标签的使用	19	5.1.2 实现 Web 端即时通信的 方法	62
3.2.2 video 标签的属性	20	5.1.3 四种 Web 即时通信技术 比较	63
3.3 Media 事件	23	5.2 短轮询原理与实现	63
3.3.1 HTML5 中 audio 和 video 的 方法	23	5.2.1 原理	63
3.3.2 HTML5 中 audio 和 video 的 属性	26		

5.2.2 实现	63	7.2 基本选择器综合使用	104
5.3 长轮询原理与实现	64	7.2.1 选择器的嵌套	104
5.3.1 原理	64	7.2.2 基本选择器优先级问题	109
5.3.2 实现	64	7.2.3 同时设定多个元素样式	110
5.4 长连接的原理与实现	66	7.3 伪类选择器	111
5.4.1 原理	66	7.3.1 状态选择器	111
5.4.2 实现	67	7.3.2 :before 和 :after	116
5.5 WebSocket 原理与前端 API	68	7.4 位置选择器	120
5.5.1 原理	68	7.4.1 :first-child	120
5.5.2 前端 API	68	7.4.2 :last-child	121
第 6 章 Canvas 绘图	69	7.4.3 :nth-child(n)	122
6.1 Canvas 基础	69	7.4.4 :nth-of-type(n)	123
6.1.1 基本 Canvas 创建	69	7.5 属性选择器	123
6.1.2 开始在 Canvas 上绘制元素	71	7.5.1 [attribute]	123
6.1.3 绘制一条直线	71	7.5.2 [attribute=value]	124
6.1.4 为直线设定样式	72	7.5.3 [attribute~value]	125
6.1.5 绘制多条直线	73	7.6 其他选择器	126
6.2 使用 Canvas 绘制图形	76	第 8 章 元素定位	127
6.2.1 绘制矩形	76	8.1 static	127
6.2.2 绘制圆形	78	8.2 relative	129
6.3 图形变换	85	8.3 absolute	130
6.4 图形的组合	89	8.4 fixed	132
6.4.1 绘制两个图形	89	8.5 float 浮动布局	133
6.4.2 新图形和原图形之间的关系	90	第 9 章 移动元素单位	137
6.4.3 加色处理	91	9.1 px	137
6.5 图像处理	91	9.1.1 需求描述一	138
6.5.1 加载图像	92	9.1.2 需求描述二	140
6.5.2 图像剪裁	94	9.2 em	143
6.5.3 像素处理	95	9.3 rem	144
第 7 章 CSS 选择器	97	9.3.1 需求描述一	145
7.1 基本选择器	97	9.3.2 需求描述二	146
7.1.1 id 选择器	97	第 10 章 盒模型	148
7.1.2 class 选择器	98	10.1 内容	149
7.1.3 元素选择器	103	10.1.1 text-align	149

10.1.2 line-height	150	14.1.1 Array 对象的属性	250
10.1.3 字体	152	14.1.2 Array 对象的方法	250
10.1.4 截断	153	14.1.3 数组遍历	254
10.2 内边距边界	155	14.2 字符串对象	254
10.3 边框边界	156	14.2.1 String 对象的属性	254
10.4 外边距边界	161	14.2.2 String 对象的方法	255
第 11 章 Flex 布局——FlexBox	162	14.3 日期	257
第 12 章 使用 CSS3 新特性	188	14.3.1 初始化一个日期	258
12.1 transform 属性	188	14.3.2 获取日期参数	258
12.2 translate 方法	189	第 15 章 JavaScript 基本语法	261
12.3 rotate 方法	194	15.1 开始使用 JavaScript	261
12.4 scale()方法	196	15.2 变量	265
12.5 skew()方法	197	15.3 基本数据类型	265
12.6 transition 方法	203	15.4 基本语句	266
12.7 帧动画	212	15.4.1 if else 语句	266
第 13 章 DOM 操作	229	15.4.2 循环语句	267
13.1 DOM 是什么	229	15.5 函数	267
13.2 通过 DOM 获取文档元素	230	15.6 对象	268
13.2.1 getElementById()方法	230	第 16 章 Window 对象	270
13.2.2 innerHTML 方法	230	16.1 setInterval()方法和 clearInterval() 方法	270
13.2.3 getElementsByTagName() 方法	233	16.2 setTimeout()方法和 clearTimeout() 方法	274
13.2.4 getElementsByClassName() 方法	234	16.3 Location 对象	277
13.3 DOM 节点	236	第 17 章 函数	279
13.3.1 节点之间的关系	236	17.1 函数的定义	279
13.3.2 节点的属性	240	17.1.1 初识函数	279
13.4 DOM 事件	245	17.1.2 匿名函数	280
13.4.1 鼠标单击事件 onclick	246	17.1.3 自我调用函数	280
13.4.2 表单改变事件 onchange	247	17.2 函数调用	281
13.4.3 完成加载事件 onload	248	17.3 函数的作用域	281
第 14 章 JavaScript 对象	249	17.4 函数的参数	283
14.1 数组	249		

第 18 章 jQuery 中的选择器.....284	第 20 章 jQuery 操纵 CSS 样式.....318
18.1 选取所有元素.....284	20.1 css()方法——直接控制元素 样式.....318
18.2 基本选择器.....288	20.1.1 获取某个元素的 CSS 属性值.....318
18.2.1 id 选择器.....288	20.1.2 设置元素的 CSS 属性.....319
18.2.2 class 选择器.....288	20.2 其他 CSS 方法.....323
18.2.3 元素选择器.....291	20.2.1 和元素尺寸相关的方法.....323
18.2.4 组合使用基本选择器.....292	20.2.2 和位置相关的方法.....324
18.3 位置选择器.....293	第 21 章 jQuery 中的动画效果.....328
18.3.1 选择特殊位置.....293	21.1 基本动画效果.....328
18.3.2 任意位置选择器.....295	21.1.1 显示和隐藏.....328
18.4 属性选择器.....297	21.1.2 淡入和淡出.....331
18.5 表单选择器.....299	21.1.3 滑动.....333
18.5.1 根据 type 选择对应的 input.....299	21.2 复杂动画效果.....335
18.5.2 状态选择器.....300	21.3 jQuery 中的动画队列.....342
18.6 灵活使用各种选择器.....301	21.4 停止动画.....344
第 19 章 jQuery 中的 DOM 方法.....303	第 22 章 jQuery 中的 AJAX.....348
19.1 获取上层元素.....303	22.1 环境搭建.....349
19.1.1 parent()方法.....303	22.2 load()方法.....350
19.1.2 parents()方法.....305	22.3 \$.get()方法和\$.post()方法.....352
19.1.3 parentsUntil() 方法.....306	22.3.1 \$.get()方法和\$.post()方法的 原理和使用.....352
19.2 获取后代元素.....307	22.3.2 实战训练.....356
19.2.1 children()方法.....307	22.4 ajax()方法.....360
19.2.2 find()方法.....309	22.5 ajaxStart()和 ajaxComplete() 方法.....364
19.3 选择兄弟节点.....310	第 23 章 jQuery Mobile 组件.....367
19.3.1 siblings()方法.....310	23.1 定义一个移动应用界面—— page 组件.....367
19.3.2 next()方法、nextAll()方法和 nextUntil()方法.....312	23.1.1 指定一个页面.....367
19.4 jQuery 与 JavaScript 的 DOM 选择 对比.....314	23.1.2 为页面划分不同的区域.....368
19.5 jQuery 中其他 DOM 节点选择 方法.....315	23.2 添加功能按钮——button 组件.....369
19.5.1 first()和 last().....315	
19.5.2 eq().....315	
19.5.3 filter()方法和 not()方法.....316	

23.2.1 按钮组件的定义方式	369	25.2.2 为输入框添加小图标	403
23.2.2 按钮组件样式设定	371	25.3 表单的种类	404
23.2.3 按钮组	373	25.3.1 搜索框	405
23.3 菜单栏	374	25.3.2 滑块输入	405
23.4 定义可滑动的面板—— panel 组件	376	25.3.3 单选框和复选框	407
23.5 弹框组件	379	第 26 章 jQuery Mobile 列表	410
23.5.1 定义基本弹框	379	26.1 简单列表	410
23.5.2 装饰弹窗	381	26.2 复杂样式的列表	414
23.6 在移动应用中添加响应式表格	385	26.2.1 气泡数字	414
23.6.1 回流表格	385	26.2.2 列表图标	417
23.6.2 列切换表格	387	26.2.3 复杂结构的列表	418
第 24 章 jQuery Mobile 的触摸事件	389	26.2.4 为列表添加功能	420
24.1 开始使用 jQuery Mobile	390	第 27 章 jQuery Mobile 项目结构	425
24.2 jQuery Mobile 事件	390	27.1 基本结构	425
24.3 触摸事件	391	27.2 页面间的过渡	427
24.3.1 tap——手指单击一次	391	27.2.1 淡入过渡——fade	427
24.3.2 taphold——手指长按 事件	393	27.2.2 翻转过渡——flip	428
24.3.3 swipe——滑动事件	394	27.2.3 抛出效果——flow	429
24.3.4 设备转动事件	396	27.2.4 滑动效果——slide	430
24.3.5 屏幕滚动事件	398	27.2.5 弹窗效果——pop	432
24.3.6 页面事件	399	第 28 章 项目实战：聊天 APP 的开发	434
第 25 章 jQuery Mobile 表单	401	28.1 移动界面编写	434
25.1 jQuery Mobile 中表单的基本 结构	401	28.1.1 聊天列表页面框架搭建	434
25.2 表单样式	402	28.1.2 聊天列表制作	435
25.2.1 隐藏标签内容	402	28.1.3 页面头部和底部的编写	437
		28.2 聊天详情页面搭建	439

第 1 章

认识 HTML

本章内容要点：

- ※ HTML 文档结构
- ※ HTML 文档构成要素以及属性和标签

学习 Web 前端开发，第一步就是学习 HTML 语言。HTML 是一个网页的骨架，HTML 中编写的内容，就是 Web 页面显示的内容。虽然 HTML 对于 Web 前端开发十分重要，但它也是非常容易学习的。本章我们先来认识 HTML 文档及其属性和标签等内容。

1.1 HTML 基本介绍

学习 HTML，主要学习的就是 HTML 的标签、属性以及 HTML 结构，本节我们先来认识这些 HTML 文档的构成要素。

1. 标签

一个页面的 HTML 部分是由各种标签组成的，一个典型的 HTML 页面结构如下：

```
<html>
  <head></head>
  <body>
    <h1>我是标题</h1>
    <p>我是段落</p>
  </body>
</html>
```

标签在 HTML 中成对出现，起始标签是由一对尖括号和英文字母组成的，如<p>，结束标签在起始标签的英文字母前面加一个/，如</p>。这样的一对标签内填写的内容就是这对标签包含的内容。<html>标签是根标签，所有页面中的其他内容都要放置在<html>的起始和终止之间。在<body>标签中，放置的是页面要显示的主要内容；<head>标签中放置一些配置信息，例如引入 CSS、JavaScript 外部脚本等。这些将在相应的章节进行介绍。

2. 属性

每一个 HTML 标签都可以指定属性，为标签指定属性的示例如下：

```
<p class="para" id="p1">段落内容</p>
```

上面的代码中，为 p 标签指定了两个属性，分别是 class 属性和 id 属性，并且为这两个属性设定了属性值。不同的标签可以设定的属性不同，但是所有标签都可以设定 class 属性和 id 属性，这两个属性和 CSS 关系紧密，是最常用的属性，将在后面进行介绍。

有些标签的可设定属性可以让这些标签有特殊的样式和功能，这些不同的内置可设定属性由 HTML 规范定义。

除此之外，也可以自定义一些 HTML 规范中没有的属性，这样做的主要目的是方便使用 Javascript 操作页面中的元素，尤其是 jQuery Mobile 框架中，频繁使用了自定义属性，这些将在后面的章节进行介绍。

3. 结构

HTML 是一个 Web 前端页面的骨架，因此 HTML 的结构很重要。HTML 主要通过标签的嵌套实现结构的搭建。例如，上面的例子就构成了一个最基本的结构，body 标签中嵌套了标题和一个段落文字内容。复杂的 HTML 结构原理也一样，在制作 Web 页面之前，先画好设计图，然后根据页面各个元素的嵌套关系进行梳理，就可以快速搭建起 Web 页面的框架。

1.2 HTML 常用标签

1. 标题

HTML 中内置了一些可以用来设置标题的标签，从<h1>一直到<h6>，这些标签内放置的文字都有预先定义好的文字样式。它们内部的文字字号比较大，并且已经进行了加粗。h1 的标题最明显，h6 的标题最小。

虽然标题标签已经预先设定好了样式，但是仍然可以通过 CSS 等手段进行自定义的样式修改。在 HTML 中，有很多和标题标签类似的标签，这些标签都有自己预先设定好的样式，但它们不是固定的，都可以借助 CSS 等手段进行修改。

```
<html>
  <head></head>
  <body>
    <h1>点击下方超链接跳转到百度</h1>
    <a href="www.baidu.com">跳转至百度</a>
```

```
</body>
</html>
```

在上面的例子中，单击“跳转至百度”，页面就会跳转到 href 属性指定的页面。

和标题标签一样，超链接标签也有预先定义好的样式，例如鼠标滑过超链接会出现下划线，单击后超链接颜色会发生变化。这些也都是可以通过 CSS 代码进行修改的。

2. 超链接

超链接标签是使用很广泛的标签，用<a>表示。超链接的起始和终止标签之间放置要现实的内容。超链接的最主要目的是跳转到链接的页面，这就需要设定超链接标签的核心属性 href。href 属性设定为一个网站链接时，当单击页面上这个标签的元素后，就会跳转到对应的链接。

3. 块元素

HTML 中最常用的是<div>元素。页面中的每一个部分都可以看成是一个 div。一般的页面都是用一系列的 div 标签构建起来的，通过 div 标签搭建页面结构。下面是使用<div>标签构建页面的简单例子。

```
<html>
  <head></head>
  <body>
    <div>
      <p>我是第一部分</p>
    </div>
    <div>
      <p>我是第二部分</p>
    </div>
  </body>
</html>
```

在上面的代码中，通过 div 元素的嵌套，将页面分成了两部分，每个部分又可以使用 div 元素继续划分。

4. 图像标签

在一个页面上引入图像的方法也很简单，只需要借助标签就可以了。img 标签有一个属性是 src，这个属性用来指定要显示的图像路径，这样就可以在页面中显示对应的图像了。

1.3 HTML 表单

表单是 HTML 中很重要的一部分。表单对应的标签是 form 元素，内部放置多个可输入的标签 input 以及提交按钮。下面是一个 form 表单的实例。

```
<from>
  姓名: <input type="text"/>
  电话: <input type="text"/>
  <input value="提交" type="submit"/>
</form>
```

上面的代码定义了一个最基础的表单，下面将对这个表单的各个元素逐一介绍。

1. form 标签

form 标签是表单的容器元素，内部通过一些输入标签组建一个 HTML 表单。form 表单有一些重要的可设置属性。首先是 method 属性，它定义了当表单提交时的提交方式，包括 GET、POST 等方法。其次是 action 属性，action 属性指定表单提交到哪个地址，也就是后台提供的对应接口。

2. input 标签

input 标签是表单内主要的内容元素。一个 input 标签对应一个输入框，用户可以在其中输入内容。input 标签的 type 属性定义了这个输入框的类型，当像上面的代码一样将 type 设置为 text 时，表示输入的基本文本内容。当 type 设置为 password 时，表示这个输入框要输入的是密码，当用户在这种 input 标签中输入内容时，网页会自动将用户输入的内容隐藏，变为黑色的圆点。当 type 设置为 submit 时，表示这是一个提交按钮，当用户单击这个按钮时，用户会自动提交这个表单到 action 指定的链接。

在 HTML5 新规范中，为 input 标签的 type 属性添加了很多其他功能丰富的属性值，可以用来建立表单中各种各样的输入框。这将在后文中详细介绍。

另一个关于 input 很重要的属性是 name 属性，这个属性是前端和后端交互的关键。input 标签设定的 name 值是什么，服务器获取到的输入标签的代号就是什么。因此在为 input 元素设定 name 属性时，一定要注意和后台同学正确对接。

1.4 HTML 和 CSS 的简单交互

HTML 的样式需要 CSS 来进行设置，二者关系非常紧密。如何使用 CSS 进行 Web 页面丰富的样式设定在后面的章节有详细介绍，这里先简单介绍一下 HTML 和 CSS 交互的基本方法。

前面说过，HTML 的大部分标签都可以设定两个属性，一个是 class 属性，另一个是 id 属性。通过这两个属性，就可以在 CSS 中为这些元素设定样式。可以为多个标签设定相同的 class 属性，然后在 CSS 中，使用特定的 CSS 语法可以获取这些 HTML 元素，并为其设定样式。id 属性的道理和 class 属性类似，只是原则上一个 id 值只能出现一次。

HTML 是搭建 Web 页面所需的最基础的知识，也是一个 Web 页面的骨架。HTML 的学习很简单，只要经常使用就能非常熟练。在下面的章节中，我们开始介绍新一代的 HTML5 标准中对基本 HTML 的强大扩展。

第 2 章

语义化标签

本章内容要点：

- ※ 语义化标签的概念
- ※ 语义化标签的使用及示例

在 HTML5 以前，HTML 中的标签只有有限的几种，诸如<div>、、<table>等，这些标签可以帮助开发者在页面中定义最基本的组件（如列表、表格等）。然而，随着 Web 开发的发展，这些基本标签已经不能满足开发者的需求。由于基本 HTML 标签种类少，开发出的 HTML 基本都是许多<div>堆砌起来的缺乏意义的文档，因此 HTML5 语义化标签应运而生。使用 HTML5 中的语义化标签构建的 HTML 文档含义更加明晰，便于页面结构的体现和后期维护。

2.1 什么是语义化标签

在 HTML5 推出之前，开发者通常使用 div 来表示页面中的各个部分，但是这些 div 没有什么实际意义（包括使用 CSS 样式的 id 和 class 等内容）。这些标签只是开发者提供给浏览器的指令，用于定义一个网页的某些部分。

以前在进行 HTML 开发时都会罗列一系列的 div 标签，导致 HTML 代码无结构性、语义混乱。其实，HTML 代码是非常重要的，不仅代表页面的结构，还要让后面的人能够通过 HTML 代码对页面的基本结构一目了然，以便于后期的开发和维护。以前开发者只能使用一系列的 div 进行 HTML 结构的编写，虽然指定了 CSS 中的 id 和 class，但是命名不一定是规范的，其他开发人员也可能看不懂之前的 HTML 结构。

从 HTML5 开始，出现了语义化标签。在语义化标签中定义了丰富的类似 div 但是各有含义的标签，用来帮助开发者更清楚地构建 HTML 结构。

HTML5 帮开发者指定了每一个语义化标签应该使用的位置、内部结构等。这些并不是 HTML 语义化标签的主要优势，下面简单说明 HTML5 语义化标签到底可以为开发者带来什么好处。

1. 使页面渲染更有意义

语义化标记为设备提供了所需的相关信息，无须开发人员考虑所有可能的显示情况（包括现有的或者将来的设备）。例如，一部手机可能会使一段标记了标题的文字以粗体显示，掌上电脑则可能会以比较大的字体来显示。无论哪种方式，一旦开发人员将文本标记为标题，就可以确信读取设备将根据自身的条件来显示合适的页面。

2. 便于团队的后期维护

如果为标签提供 CSS 中的 id 和 class，很可能会出现不规范的情况。当开发者使用语义化标签后，由于语义化标签是规范的，因此方便了后期维护。

3. 使 HTML 结构更加清晰

使用 HTML 语义化标签可以让页面的结构更加清晰，按照规范定义的页面，便于大家快速了解每一个结构的功能和作用。

4. 为搜索提供帮助

SEO 优化等语义化标签可以给搜索引擎一个更清晰、更好的搜索信号，让页面更容易被搜索。

当然，有了语义化标签，并不代表不需要 div 了，div 仍然是很重要的。此外，语义化标签基本不会为开发者的页面添加样式，还需要使用 CSS 代码对样式进行控制。因此，要明确 HTML5 语义化标签的目的，即让页面的结构更加清晰、更加便于阅读和解析。

下面我们将逐一介绍 HTML5 为开发者提供的语义化标签。

2.2 header 标签

header 标签主要用来定义页面的页眉，如标题等。如果开发者想为自己的页面添加一个导航栏，就可以使用 header 这个标签。

【示例】 使用 header 标签的代码如下：

```
<!DOCTYPE>
<html>
<head>
<meta charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1"/>
<style type="text/css">
</style>
</head>
```



```

<body>
  <header>
    <hgroup>
      <h1>网站标题</h1>
      <h1>网站副标题</h1>
    </hgroup>
  </header>
</body>
</html>

```

使用 div 的 HTML 结构如下:

```

<!DOCTYPE>
<html>
<head>
<meta charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1"/>
<style type="text/css">
</style>
</head>
<body>
  <div class="header">
    <div class="top">
      <ul>
        <li>导航</li>
        <li>导航</li>
        <li>导航</li>
        <li>导航</li>
        <li>导航</li>
      </ul>
    </div>
    <div class="nav">
      <input type="search" placeholder="输入内容">
    </div>
  </div>
</body>
</html>

```

对比一下,可以看出使用 div 的页面结构十分混乱,并且需要开发者自己设定元素的 class 等,以表明其含义。现在很多网站都流行两个导航栏,使用 div 的时候,如果想定义这样的结构,就需要用不同 class 的 div,比如上面的 nav 和 top,显然会让人误解。如果使用语义化标签,那么使用两个<hgroup>即可将网站的导航栏分开,无疑会让页面的结构更加清晰。